

Vysoká škola báňská – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

VoIP klient pro operační systém Windows Phone

VoIP client for Windows Phone operating system

Zadání bakalářské práce

Student:

Jakub Prášil

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R059 Mobilní technologie

Téma:

VoIP klient pro operační systém Windows Phone
VoIP Client for Windows Phone Operating System

Zásady pro vypracování:

1. Navrhněte a vytvořte VoIP klienta pro operační systém Windows Phone.
2. Klient bude vytvořen na základě již existujících klientů s přidanou podporou pro DTMF volbu během video hovoru, bude taktéž přidána podpora pro textovou komunikaci.
3. Klient bude podporovat audio, video přenosy a textovou komunikaci.
4. Vytvořená aplikace bude podporovat SIP, H.263, H.264, DTMF, SOAP, zprávy přes SIP protokol a XMPP protokol.
5. Student vytvoří dokumentaci k vytvořené aplikaci.

Seznam doporučené odborné literatury:

M. Voznak, Voice over IP. VSB-Technical University of Ostrava: College book, 1st. ed., Ostrava, 2008.

Meggelen, J., P., Smith, J., Madsen, L.: Asterisk: The Future of Telephony. O'Reilly Media

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

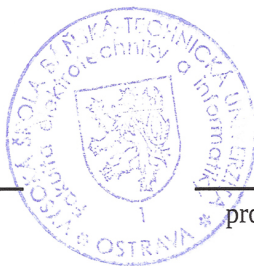
Vedoucí bakalářské práce: **Ing. Lukáš Kapičák**

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2014



doc. Ing. Miroslav Vozňák, Ph.D.
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

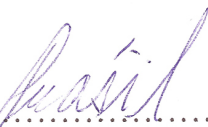
Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 *Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava*.

V Ostravě 23. dubna 2014

........

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 23. dubna 2014

........

Rád bych poděkoval pánovi Ing. Lukášovi Kapičákovi za odbornou pomoc, cenné rady a konzultaci při vytváření této bakalářské práce.

Abstrakt

Téma bakalářské práce je vývoj aplikace pro operační systém Windows Phone v aktuální verzi 8. Konkrétně je to aplikace zajišťující komunikaci přes IP počítačovou síť. Přibližně první polovina práce se zabývá teorií popisující operační systém, vývojové prostředí pro aplikace a hlavně technologie využité při tvorbě praktické části. Popisují se zde hlavně technologie, které úzce souvisí s vytvořením aplikace. To znamená VoIP, signalizační protokol SIP včetně jeho předchůdce H.323, XMPP protokol pro zprávy. Dále je zde zmínka také o tónové volbě DTMF a SOAP. V druhé polovině práce popisují praktickou část práce, tudíž představení aplikace. Zde je zamření na knihovny použité v aplikaci a strukturu aplikace. Poté popisují aplikaci z pohledu uživatele, její možnosti a nastavení.

Klíčová slova: Windows Phone, VoIP, H.323, SIP, XMPP, DTMF, SOAP

Abstract

The bachelor thesis is the development of application for Windows Phone in the current Version 8. Specifically, it is application for communication over IP computer network. Approximately the first half of the work is occupied with the theory describing the operating system, development environment for applications and technology mainly used in the creation of a practical part. Described herein the main technologies that are closely related with the creation of application. It means VoIP, SIP signaling protocol including his predecessor H.323, XMPP protocol for messaging. Then there is also mention of DTMF and SOAP. In the second half I describe the practical part of the work, so presentation of application. here is focus to the libraries used in the application and the application framework. After I describe the application with a user's perspective, its options and settings.

Keywords: Windows Phone, VoIP, H.323, SIP, XMPP, DTMF, SOAP

Seznam použitých zkratk a symbolů

ACK	– Acknowledgement
CD	– Compact Disc
DTMF	– Dual-Tone Multi-Frequency
HTTP	– Hypertext Transfer Protocol
IETF	– Internet Engineering Task Force
IP	– Internet Protocol
ISDN	– Integrated Services Digital Network
ITU	– International Telecommunication Union
LTE	– Long Term Evolution
MCU	– Multipoint Control Unit
MOS	– Mean Opinion Score
OMA	– Open Mobile Alliance
OS	– Operating System
OSI	– Open Systems Interconnection
PSTN	– Public Switched Telephone Network
PTT	– Push to talk
QoS	– Quality of Service
RFC	– Request For Comments
RCS	– Rich Communication Services
RPS	– Remote procedure call
RTCP	– RTP Control Protocol
RTP	– Real-time Transport Protocol
SDK	– Software Development Kit
SDP	– Session Description Protocol
SMTP	– Simple Mail Transfer Protocol
SIP	– Session Initiation Protocol
SOAP	– Simple Object Access Protocol
TCP	– Transmission Control Protocol
UDP	– User Datagram Protocol
VoIP	– Voice over Internet Protocol
VPN	– Virtual Private Network
XAML	– Extensible Application Markup Language
XEP	– XMPP Extensions

XML	– Extensible Markup Language
XMPP	– Extensible Messaging and Presence Protocol
XS	– XMPP Standards Foundation

Obsah

1	Úvod	6
2	Přenos dat v telefonní síti	7
2.1	Sítě s přepínáním okruhů	7
2.2	Sítě s přepínáním paketů	7
3	VoIP	8
3.1	Princip hovoru	8
3.2	Protokoly	9
4	Signalizační protokoly VoIP	10
4.1	H.323	10
4.1.1	Architektura standardu H.323	10
4.1.2	Výhody H.323	10
4.2	SIP	11
4.2.1	Prvky architektury	11
4.2.2	Metody	11
4.2.3	Odpovědi	12
4.2.4	Hlavička SIP	12
4.2.5	Pojmy	12
4.2.6	Vyjednání spojení	13
5	Komunikační protokoly VoIP	14
5.1	RTP protokol	14
5.2	RTCP protokol	14
5.2.1	RTCP služby	14
5.2.2	RTCP zprávy	14
6	Kodeky	15
6.1	Typy kodeků	15
6.2	Kvalita kodeku	15
6.3	Použité kodeky	15
6.4	H.263	16
6.5	H.264	16
7	Textový protokol XMPP	17
7.1	Identifikátor uživatele	17
7.2	Komunikace v XMPP	17
7.3	Vlastnosti XMPP	17
8	DTMF	18
8.1	Vytvoření tónu	18

9 SOAP	19
9.1 Zpráva	19
10 Windows Phone	20
10.1 Platformy	20
10.1.1 Windows CE	20
10.1.2 Windows NT	20
10.2 Verze Windows Phone	20
10.2.1 Vývoj verzí	20
10.3 Architektura Windows Phone 8	21
10.4 Aplikační modely	22
10.5 Grafické rozhraní	22
10.6 Aplikace na pozadí systému	22
11 Vývoj aplikací	24
11.1 Nástroje pro vývoj	24
11.1.1 Windows Phone SDK	24
11.1.2 Představení vývojového prostředí	24
11.2 Typy aplikací	25
12 Aplikace VOIPtalk	26
12.1 Použité frameworky a knihovny	26
12.1.1 MatriX XMPP	26
12.1.2 Doubango	27
12.2 Struktura aplikace	27
12.3 Uživatelské rozhraní	28
12.4 Uživatelská dokumentace	29
12.4.1 Ovládání aplikace	29
12.4.2 Nastavení aplikace	32
12.4.3 Ovládání hovoru	36
13 Testování aplikace	37
13.1 Apikační testování	37
13.2 Testování zpráv	37
13.2.1 Server Openfire	37
13.2.2 Server Jabbim.cz	37
13.3 Testování SIP spojení	37
13.3.1 Spojení dvou klientů	38
14 Závěr	39
15 Literatura	40
Přílohy	40

Seznam tabulek

8.1	Frekvence DTMF.	18
11.1	Typy aplikací.	25

Seznam obrázků

3.1	Převod hlasu ve VoIP.	8
4.1	Vyjednání spojení SIP.	13
10.1	Architektura Windows Phone 8.	21
10.2	Aplikační model Windows Phone 8.	22
11.1	Nástroj Visual Studio určený k tvorbě aplikací pro Windows Phone.	24
12.1	Screen aplikace s dialogem knihovny MatriX XMPP.	26
12.2	Struktura aplikace.	27
12.3	Schéma rozložení panoramatické obrazovky.	28
12.4	Screen po spuštění aplikace.	29
12.5	Screen aplikace zobrazující práci s kontakty.	30
12.6	Screen aplikace zobrazující XMPP zprávy.	31
12.7	Screen aplikace s nastavením identity.	32
12.8	Screen aplikace s nastavením sítě.	33
12.9	Screen aplikace s nastavením XMPP zpráv.	34
12.10	Screen aplikace s nastavením zamykání obrazovky.	35
12.11	Screen aplikace zobrazující přijetí a průběh hovoru.	36
13.1	Spojení dvou koncových zařízení pomocí ústředny Asterisk.	38

1 Úvod

V dnešní době s rozšířením internetu, kdy disponuje datovým připojením téměř každé druhé mobilní zařízení, se technologie VoIP stává nedílnou součástí. VoIP je moderním trendem v oblasti telekomunikací. Zajišťuje hovorovou komunikaci v rámci datových IP sítí. V součinnosti s novodobým operačním systémem Windows Phone, který zaznamenává na trhu mobilních zařízení překvapivý vzestup, se jedná o konkurenci schopnou možnost nahrazení klasické telefonie. Z tohoto důvodu jsem si jako téma práce zvolil vývoj VoIP aplikace pro systém Windows Phone. Technologie mě láká vzhledem k tomu, že se jí doposud vývojáři nevěnují natolik, jako konkurenčnímu systému Android. Platforma poskytuje široké možnosti vývoje.

Mnou vytvořená aplikace pracuje v součinnosti s protokolem SIP, který zajišťuje relaci pro audio i video hovory. Dále aplikace uživateli poskytuje možnost textové komunikace prostřednictvím XMPP protokolu.

První část práce se věnuje technologiím použitých v aplikaci. Druhá část na teoretické poznatky navazuje a představuje vytvořenou aplikaci.

Druhá kapitola se zabývá základními způsoby přenosu dat v telefonních sítích a jejich rozdíly. Shrnuje sítě s principem přepínáním okruhů a přepínáním paketů.

Třetí kapitola se věnuje technologii VoIP. Popisuje přenos hovorových dat přes počítačovou síť. Následuje kapitola popisující dva nejčastěji používané signalizační protokoly. V mé aplikaci slouží pro signalizaci SIP protokol, proto je problematika protokolu detailněji rozebrána. Na signalizační protokoly navazuje kapitola, která vysvětluje pojem komunikační protokoly.

Šestá kapitola řeší snížení objemu přenášených informací použitím kódovacích algoritmů. Zabývá se funkcemi, rozdíly a důvody použití kodeků.

Následuje kapitola sedmá, která se zabývá protokolem XMPP. Protokol je důležitou součástí aplikace pro textovou komunikaci. Kapitola popisuje postup komunikace a vlastnosti protokolu.

Další kapitoly pojednávají o tónové volbě DTMF, a také o SOAP protokolu a jeho využití při komunikaci mezi službami.

Desátá kapitola je věnována operačnímu systému Windows Phone. V kapitole popisují rozdíly mezi verzemi, architekturu systému, aplikační modely, grafické rozhraní a problematiku spouštění procesů na pozadí systému. Následuje kapitola, která popisuje nástroje určené pro vývoj aplikací tohoto systému.

Předposlední kapitola se zabývá vytvořenou aplikací. Zde je uveden použitý framework Doubango poskytující aplikaci rozhraní pro hovorovou komunikaci a knihovna MatriX XMPP, která rozšiřuje aplikaci o textovou komunikaci. Dále v kapitole uvádím strukturu aplikace, její ovládání a potřebné nastavení.

Poslední kapitola shrnuje průběh testování vytvořené aplikace a prostředky použité pro testování použity.

2 Přenos dat v telefonní síti

2.1 Síť s přepínáním okruhů

Přepojování okruhů pochází ze starších standardních telefonních sítí. V moderní komunikaci se tento princip zřídka využívá v sítích s vytáčeným spojením.

Přenos dat je založen na vyhrazení kanálu mezi odesílatelem a příjemcem, kdy je tento kanál určen pouze jednomu spojení po dobu celého přenosu. Ústředny na lince mezi spojenými klienty obstarávají poskytnutí vhodného přenosového kanálu. V případě, že některá z ústředen nebude moci poskytnout vhodný kanál, je odesílateli zaslán signál o obsazenosti linky.

Nevýhodou je nízká efektivita využití přenosového pásma, jelikož nevyužitý prostor v komunikaci mezi dvěma klienty nemůže být využit pro jiné spojení. Problém nevyužití přenosového pásma částečně řeší multiplexování, které slučuje více signálů do jednoho signálu. Výhodou těchto sítí je nekolísající kvalita přenášených dat v závislosti na šířce pásma.

2.2 Síť s přepínáním paketů

Paketové síť rozdělují data do mnoha částí (paketů), které jsou postupně odesílány k příjemci. Pakety nemají předem určenou trasu, proto bývá trasa jednotlivých paketů mnohdy odlišná v závislosti na využití sítě. Spolu s daty jsou v paketech obsažena data důležitá pro komunikaci, jako jsou adresy komunikujících zařízení, kontrolní součty atd. Paket dorazí na zařízení na síti, které vyčte z hlavičky paketu adresu příjemce. Podle adresy se zařízení rozhodne, kterou cestou paket odešle. Pakety se sestavují, až když dorazí k příjemci. Výsledkem jsou data v původní podobě. Hlavní výhodou je efektivnější přenos dat z důvodu úspory přenosové šířky pásma. Nevýhoda je kolísání kvality hovorů zapříčiněnou ztrátou paketů.

3 VoIP

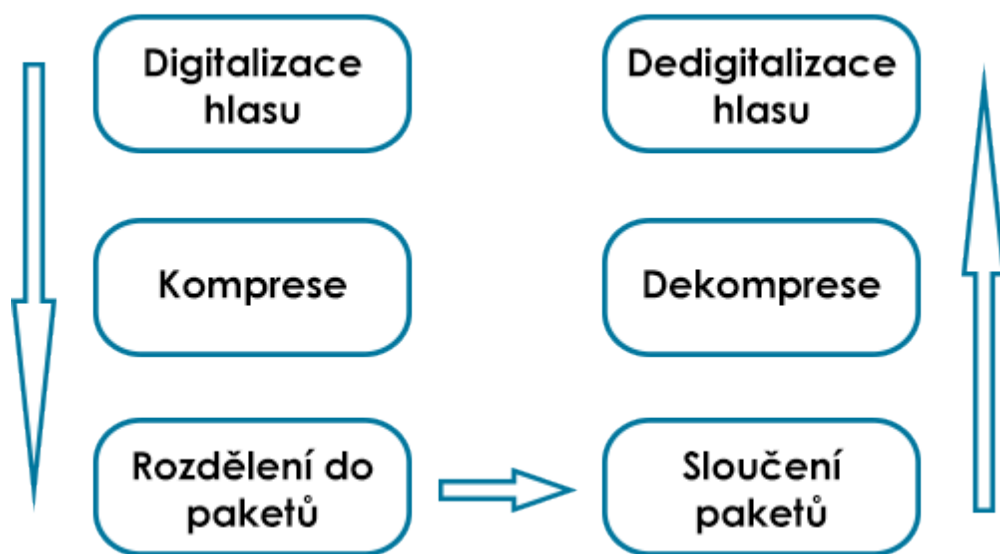
Slouží jako alternativa klasické telefonní linky s přepojováním okruhů. Tato moderní technologie využívá IP protokolu třetí vrstvy OSI modelu k přenosu hlasu v digitální podobě přes počítačovou síť, případně jinou datovou síť [8]. Je založena na síti s přepojováním paketů, proto je nutnou podmínkou zajistit potřebnou kvalitu služby (QoS). QoS zajišťuje řízení datových toků a vyhrazení přenosové kapacity linky.

Hovorová data jsou přenášena na transportní vrstvě OSI modelu ve formě UDP nebo TCP paketů. Častěji využívaný je nespojově orientovaný protokol UDP, jelikož umožňuje komunikaci v reálném čase. Pátou vrstvu OSI modelu zastává RTP protokol, který obsahuje část hovorových dat. Obvykle se jedná o 20 a 30 ms fragmenty.

Pro snížení využití sítě se využívají kódovací algoritmy (kodeky), které minimalizují velikost přenášených dat.

3.1 Princip hovoru

Hlas je převeden z analogové do digitální podoby, následně je provedena komprese dat pomocí kódovacích algoritmů (kodeků) a rozdělení do jednotlivých datových paketů. Pakety jsou odesílány přes IP síť. Po přijetí paketů proběhne sloučení paketů, dekomprese a dedigitalizace.



Obrázek 3.1: Převod hlasu ve VoIP.

V tradiční telefonii, kdy je hlas tvořen signálem (analogovým / digitálním) se k signálu připojují různé šумы. Tyto šумы zabraňují srozumitelnosti hovoru na vyšších vzdálenostech. Při zesilování signálu se s hlasem zesilují také šумы. V IP telefonii se hlas přenáší bitově, proto zde nevznikají šумы. Neobjevuje se ztráta kvality přenášených dat v případě

3 VOIP

kopírování bitů při spojení na vyšší vzdálenosti. Další rozdíl je v propustnosti pásma pro přenášení hlasu. V tradiční digitální lince je potřeba přenosová rychlost pásma 64 kbps, ovšem ve VoIP technologii při využití vhodných kodeků postačuje pásmo s rychlostí 6 kbps.

3.2 Protokoly

Protokoly technologie VoIP se dělí do dvou skupin:

Signalizační – využívají se pro návázání, řízení a ukončení spojení. Mezi nejznámější protokoly patří H.323 a SIP.

Komunikační – určeny pro přenos hlasu. Protokoly RTP a RTCP.

4 Signalizační protokoly VoIP

Signalizační protokoly jsou podstatným prvkem této technologie. Slouží pro navázání, řízení a ukončení spojení. Poskytují zařízením různých výrobců informace potřebné pro řízení komunikace. Základ technologie VoIP tvoří dvojice protokolů H.323 a SIP. Starší velice rozšířený protokol je H.323, v dnešní době již na ústupu a nahrazuje ho zjednodušený protokol SIP.

Hlavním rozdílem mezi těmito protokoly je struktura dat, zatímco H.323 je založeno na binárně orientovaných datech, jeho nástupce SIP využívá textovou strukturu dat.

4.1 H.323

Je standardem mezinárodní telekomunikační unie ITU-T, která zajišťuje jeho distribuci [4]. Jedná se o souhrn doporučení, jejichž úkolem je navázání relace pro audio i video přenos. Definuje komunikaci mezi terminály, zařízeními a multimediálními službami na sítích s přepojováním paketů.

4.1.1 Architektura standardu H.323

Terminál je označení pro koncová zařízení (například VoIP telefony, počítačový software).

MCU zajišťuje konferenčního spojení, to znamená 3 a více uživatelů najednou.

Gateway (Brána) umožňuje komunikaci se sítěmi, které nepodporují protokol H.323. Jedná se o sítě ISDN a PSTN

Gatekeeper (Správce) je pro danou administrativní část řídicím prvkem, který zajišťuje proces autorizace, autentifikace, překlad adres a řízení přenosové kapacity.

4.1.2 Výhody H.323

- nezávislost na přenosové síti
- kompatibilita koncových zařízení
- možnost spojení různých platforem
- konferenční hovory
- řízení zatížení sítě, dle šířky přenosového pásma

4.2 SIP

Signalizační protokol, jehož úkolem je navázat, řídit a ukončit spojení koncových zařízení neboli klientů. Koncepčně podobný protokolům HTTP pro hypertextové dokumenty a SMTP pro elektronickou poštu. Struktura dat je reprezentována textově a skládá se z elementů typu <název>:<hodnota>. V těchto elementech jsou obsaženy informace o komunikaci. Z důvodu jednoduché koncepce postupně nahrazuje svého předchůdce H.323. SIP stará pouze o relaci mezi klienty a o samotnou komunikaci se nestará, kdežto H.323 dohlíží na celou komunikaci. Vlastnosti relace jsou prezentovány protokolem SDP, kdežto pro přenášená data se využívá protokol RTP. Komunikace je založena na principu klient-server. Klient zadá požadavek serveru na spojení a server zajistí vyjednání tohoto spojení.

Výhodou SIP protokolu je možnost zprostředkování textové komunikace v reálném čase. To zajišťuje rozšíření o standard SIMPLE.

4.2.1 Prvky architektury

User Agent (UA) Jedná se o zařízení sítě standardu SIP. Rozlišují se na UAC (User agent client) a UAS (User agent server). Klient zašle inicializaci spojení a server na jeho žádost reaguje.

SIP server Zajišťuje spojení mezi výše uvedenými volajícími (UA).

4.2.1.1 Typy serverů

Register – Ukládá registrované klienty do databáze lokalizační služby, kde uchovává klienty dané domény.

Proxy – Je v roli prostředníka, který zajišťuje spojení mezi klienty.

Redirect – Zajišťuje přesměrování klienta na jiný server.

Location – Lokalizační databáze Proxy a Redirect serverů.

4.2.2 Metody

Slouží pro obsluhu relace SIP sítě. Základní metody dle RFC 3261 [5]

INVITE – Požadavek na sestavení spojení.

ACK – Přijetí požadavku.

BYE – Ukončení sestaveného spojení.

CANCEL – Ukončení ještě nesestaveného spojení.

REGISTER – Požadavek pro registraci nebo odregistrování

OPTIONS – Požadavek na server, který se dotazuje na jeho vlastnosti.

4 SIGNALIZAČNÍ PROTOKOLY VOIP

4.2.3 Odpovědi

Jedná se o odpovědi na výše uvedené požadavky. Informují o úspěchu či neúspěchu požadavku.

1xx – informační zpráva

2xx – úspěšný požadavek

3xx - přesměrování

4xx – chyba na straně klienta, to znamená nekorektně položený dotaz

5xx – chyba na straně serveru

6xx – globální selhání, server není schopen požadavek vykonat

Poznámka 4.1 V praxi *xx* reprezentují čísla konkrétní odpovědi. Například 180 Ringing nebo 200 OK.

4.2.4 Hlavička SIP

Podstatné položky hlavičky zprávy. Některé položky hlavičky jsou podobné jako v protokolu SMTP.

From Adresa volajícího klienta.

To Adresa volaného klienta.

Via Adresa serveru nebo klienta, který odeslal požadavek. Počet záznamů Via se odvíjí od počtu serverů, kterými zpráva prošla.

Call-ID Identifikace spojení.

Contact Aktuální klientská adresa.

4.2.5 Pojmy

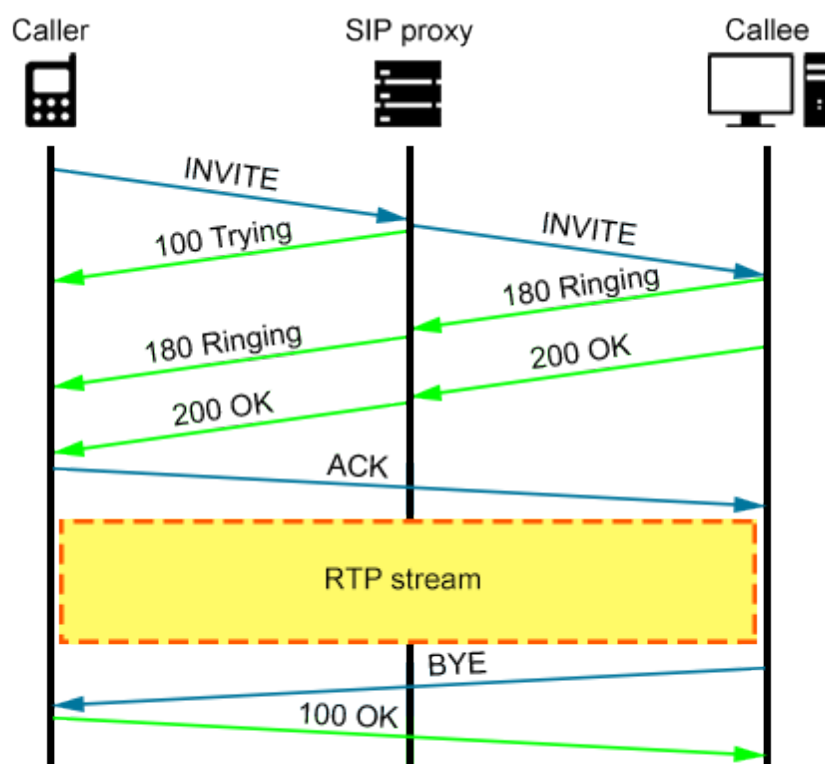
SIP adresa Jednoznačně identifikuje uživatele při SIP komunikaci. Struktura je stejná jako u e-mailu, navíc obsahuje předponu `sip:`. Obecná adresa `sip:klient@doména`, klient představuje název nebo číslo uživatele a doména zde zastává IP adresu, nebo doménový název.

Transkace Obsahuje požadavek a příslušnou odpověď na něj.

4.2.6 Vyjednání spojení

Jiným slovem signalizace, je to proces, při kterém se navazuje spojení UA. Spojení je navázáno obdobně jako v three-way-handshake známého z TCP.

Volající klient odešle požadavek INVITE, který obsahuje informace o nabízeném spojení. Pokud požadavek dorazil až k volanému klientovi a ten je schopný jej uskutečnit, vrací se odpověď o provádění požadavku. Například, že telefon volaného začal vyzvánět (RINGING). Jestliže volaný spojení přijme je vyslána odpověď o přijetí spojení (OK), volající strana tuto odpověď potvrdí (ACK) a relace je navázána. V případě ukončení spojení, vyšle jedna z komunikujících stran požadavek BYE. Požadavek BYE je následně potvrzen (ACK) protistranou.



Obrázek 4.1: Vyjednání spojení SIP.

5 Komunikační protokoly VoIP

5.1 RTP protokol

Protokol standardizovaný normou RFC 3550 společnosti IETF [7]. Určený k přenosu multimedií v reálném čase. Zastoupení v streamových (proudových) přenosech, jako jsou hovorové sítě, konference a PTT systémy. Data jsou rozdělena a přenášeny uvnitř UDP paketů. Z důvodu rozdělení přenášených dat, každý paket obsahuje sekvenční číslo, které slouží pro sestavení dat v přijímacím procesu. Dále hlavička protokolu obsahuje informace o typu přenášených dat, synchronizaci, kódování a časová značka.

5.2 RTCP protokol

Stejně jako protokol RTP je standardizován normou RFC 3550 [7]. Protokol je určený k řízení datových toků RTP protokolu. Pro řízení výkonnosti a diagnostiku využívá periodického vysílání paketů od všech účastníků RTP relace [8].

5.2.1 RTCP služby

- Řídí intervaly vysílání RTCP
- Identifikuje zdroje RTP
- Přenáší minimální informace řízení relace
- Předává aplikaci informace o kvalitě dat

5.2.2 RTCP zprávy

RTCP umožňuje výměnu zpráv o právě probíhající konferenci. Pro přenos zpráv se využívá UDP protokolu s jiným portem, než využívá protokol RTP. Standard obsahuje následující zprávy.

Sender Report (SR) – Odesílány aktivními účastníky konference v pravidelných intervalech. Zpráva obsahuje informace o komunikaci a přijímací statistiky RTP paketů

Receiver Report (BR) – Odesílány pasivními účastníky konference. Obsahuje informace o kvalitě služby, zpoždění a problémech příjemců pro odesílatele

Source Description Message (SDS) – Zprávy zasílané v pravidelných intervalech. Obsahují informace o vysílači dat, který zprávu odesílá.

Application-Specific Message (APP) – Umožňuje definici nových zpráv mimo zprávy standardu.

Goodbye Message (BYE) – Oznámení vysílače dat o ukončení konference.

6 Kodeky

Jiným slovem algoritmy, slouží ke kompresi a dekompresi přenášených dat. Přenášenými daty rozumíme signál nebo datový proud (stream). Cílem kodeků je snížit počet bitů k uchování informací a zároveň zachování kvality vstupních dat. Předpokládá se minimální rozdíl mezi vstupními a výstupními vzorky. Veškeré tyto vlastnosti závisí na možnostech zvoleného kodeku.

6.1 Typy kodeků

Waveform kodeky – Typické pro tyto kodeky je vysoký bitrate (počet bitů přenesených za jednu sekundu) a jejich nízká složitost. Velice kvalitní projev u rychlostí nad 16 kbit/s, pod touto hranicí se kvalita rychle ztrácí.

Source kodeky – Jsou opakem waveform kodeků, tudíž pracují s nízkým bitratem. Hlas je rozdělen a vyjádřen matematicky, proto je výsledný projev velice umělý.

Hybridní kodeky – Jedná se kombinace waveform a source kodeků. Z jejich sloučení vyplývá, že kodek má průměrnou kvalitu výstupních dat s použitím střední hodnoty bitratu.

6.2 Kvalita kodeku

Jedním z faktorů určení kvality kodeků je MOS. Rozděluje se na tři metody rozpoznání kvality hovoru (objektivní, subjektivní a výpočetní E-model) [6].

Hodnoty jsou v rozmezí 1-5. Minimální 1 (špatná, velmi nesrozumitelná) a maximální 5 (vynikající, zhoršení není znatelné).

MOS 1 – kvalita velmi špatná, hovor nesrozumitelný.

MOS 2 – kvalita špatná, hovor mírně nesrozumitelný.

MOS 3 – ucházející kvalita, podstatné zhoršení.

MOS 4 – dobrá kvalita, mírné zhoršení.

MOS 5 – vynikající kvalita, zhoršení není patrné.

Hodnota 3 je považována jako hraniční pro telekomunikace.

6.3 Použité kodeky

Kodeky využívané v aplikaci jsou následující:

- Audio
 - G.711
 - GSM

- iLBC
 - Speex
 - OPUS
 - G.722
 - G.729
- Video
 - VP8
 - MP4V-ES
 - Theora
 - H.264
 - H.263

Poznámka 6.1 V zadání jsou přesně specifikovány pouze dva video kodeky (H.263 a H.264), proto dále popisují pouze tyto dva standardy.

6.4 H.263

Kodek byl vytvořen na základech svého předchůdce H.261 spolu se standardy MPEG-1 a MPEG-2. Prvotní využití získal při videokonferencích. Později našel využití ve flashových video webových aplikacích. Následují rozšířené verze H.263+ a H.263++.

6.5 H.264

Nástupce kodeku H.263, využívá standardu MPEG-4. Z důvodu vysoké míry komprese je v dnešní době asi nejpoužívanějším kodekem. Oproti svému předchůdci dosahuje až dvojnásobně vyšší komprese se zachováním velice kvalitního obrazu. Pro přenos to znamená menší objem dat ke zpracování.

7 Textový protokol XMPP

Xmpp je založený na značkovacím jazyce XML. Slouží pro zpracování IM komunikace a zjišťování stavu ostatních uživatelů. Původně byl určen pro IM komunikaci v síti Jabber, proto byl dříve známý spíše pod tímto názvem. [9].

Spojení je decentralizované typu klient–server. V síti neexistuje žádný centrální server pro připojení všech klientů. Každý klient se připojuje pomocí jména a hesla pouze k serveru, ke kterému je registrován, ten jako jediný má uloženou identitu klienta.

7.1 Identifikátor uživatele

Jednoznačný identifikátor uživatele, syntakticky podobný e-mailovým adresám. Identifikátor je ve tvaru `název-uživatele@server`, může být upřesněn pomocí /upřesnění. Často je označován jako JID, neboli Jabber ID. Na jednom uživatelském účtu může být připojeno více zařízení, rozhoduje se mezi nimi podle priority nebo celé identifikační adresy.

7.2 Komunikace v XMPP

Mezi klienty se nevytváří relace, klient komunikuje pouze se svým serverem na portu 5222. Server přijme zprávu a převezme informace, komu zprávu doručit. Zpráva je doručena jakmile se daný klient k serveru přihlásí. Uživatelé jednoho serveru mohou komunikovat také s uživateli ostatních serverů. V tomto případě mezi sebou komunikují servery na portu 5269 a předávají si informace potřebné ke komunikaci.

Servery pokrývají poměrně malou oblast klientů. Jedná o otevřenou technologii, tudíž kdokoli může administrovat vlastní server.

7.3 Vlastnosti XMPP

Jak již bylo zmíněno, tato technologie je otevřená a decentralizovaná. Za důležitou vlastnost XMPP se považuje, že je rozšiřitelný nad rámec RFC norem pomocí XEP standardů, dnes jich existuje okolo dvou set. Jsou vytvářeny nadací XSF, která tyto standardy vytváří na základě zkušeností a testování [10].

Považuje se za poměrně bezpečný přenos zpráv, jelikož je komunikace rozprostřena na mnoho serverů a servery mohou být ukryty i uvnitř lokální sítě. Možnost využití jako víceuživatelský chat, přenos dat, komunikace mezi programy, řízení různých služeb a procesů.

Identifikační údaje jsou uloženy na serveru, nikoliv na lokálním zařízení. To umožňuje připojení na server z mnoha různých míst.

8 DTMF

DTMF neboli tónová či kmitočtová volba je základem řízení hlasové služby. Umožňuje ovládat ústředny a měnit jejich nastavení. Reprezentuje stisk klávesy jako krátké písknutí. Toto písknutí je generováno pomocí dvou sinusových signálů o přesné frekvenci. Signály jsou vždy jeden s nižší frekvencí a druhý o vyšší frekvenci. Těchto frekvencí je celkově 8 v rozmezí 697 – 1 633 Hz. Rozmezí je zvoleno v závislosti na frekvenčních filtrech a telekomunikačních sítích, které pracují s rozsahem 300 – 3 400 kHz. Standardně se přenáší 50 ms tónu a 50 ms pauza mezi tóny.

8.1 Vytvoření tónu

Tabulka zobrazuje schéma pro vytváření jednotlivých signálů, vždy se sečtou amplitudy dvou příslušných frekvencí. Při stisku klávesy 1, se kombinují frekvence 1209 Hz a 697 Hz. Podle normy je přípustná odchylka maximálně 1,5% od uvedené hodnoty. Průběhy obou frekvencí jsou sinusové o přibližně stejné amplitudě. Amplituda vyšší frekvence může dosahovat hodnoty maximálně o 4 dB vyšší. V tabulce jsou vyobrazeny také frekvence pro tónovou volbu písmem A – D. Ty jsou použity výjimečně, slouží jako systémové tóny pro nastavení ústředěn [11].

	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

Tabulka 8.1: Frekvence DTMF.

Poznámka 8.1 Aplikace obsahuje možnost při hovoru zadávat čísla na standardním číselníku. Při stisku klávesy aplikace vyše ústředně daný tón, která tón dále zpracuje.

9 SOAP

Je protokolem webových služeb, slouží pro výměnu XML zpráv mezi webovými aplikacemi, tudíž se jedná o komunikaci typu peer-to-peer. Zprávou rozumíme jednostranný přenos, který obsahuje informace aplikací. Pro přenos se využívá protokolu HTTP nebo SMTP. Častěji je použit protokol HTTP, z důvodu rozšířenosti v oblasti internetu ??.

SOAP je použit jako náhrada RPC, který komunikuje na bázi požadavku a odpovědi. Při využití tohoto modelu je do komunikace zařazen webový server. V případě kdy server přijme požadavek spustí webovou službu, které požadavek předá. Odpověď zaslaná zpět klientovi obsahuje výsledek webové služby.

9.1 Zpráva

Zpráva je ve formátu XML. Tento formát je vhodný pro komunikaci z důvodu své variability, a také vhodné přehlednosti jak pro lidský faktor, tak pro zařízení.

Hlavní element zprávy je Envelope. Uvnitř elementu se skrývají elementy Header a Body. Header není povinný, přenáší se zde pomocné informace, například autentizace. Element Body je naopak povinný, přenáší se v něm informace identifikující proces a parametry, které proces předává.

Poznámka 9.1 SOAP byl určen pro mou aplikaci, jako možnost snadného nastavení ústředny. Po dohodě s vedoucím bylo od tohoto řešení upuštěno.

10 Windows Phone

Windows Phone je operační systém pro mobilní zařízení, vyvinutý společností Microsoft. Byl nasazen jako nástupce systému Windows Mobile. První verze byla vydána v druhé polovině roku 2010 [2]. Existuje ve dvou základních verzích, Windows Phone 7 a aktuální verze Windows Phone 8. Z důvodu rozličné platformy nejsou vzájemně kompatibilní, verze 7 je založena na omezeném systému Windows CE, avšak verze 8 je založena na přizpůsobeném jádře Windows NT, který byl doposud využíván pouze v počítačích.

10.1 Platformy

10.1.1 Windows CE

Samostatný systém určený pro malé a mobilní zařízení, založený na hybridním jádře. Je open source (volně dostupný) a rozšiřitelný, což umožňuje následné přizpůsobení pro dané zařízení. Z důvodu deterministického plánování procesů se jedná o operační systém reálného času nenáročný na paměť zařízení.

Procesory: MIPS, ARM, Intel x86.

10.1.2 Windows NT

NT platformu využívá Microsoft od počátků svých operačních systémů. Výhodou systému je preemptivní multitasking, zajišťující bezpečnost při použití aplikací, které by mohly ohrozit systém nekorektním vývojem. Pro potřeby Windows Phone verze 8 byla upravena od stávajícího desktopového operačního systému Windows 8.

10.2 Verze Windows Phone

- Windows Phone 7
- Windows Phone 7.5
- Windows Phone 7.8
- Windows Phone 8
- Windows Phone 8.1

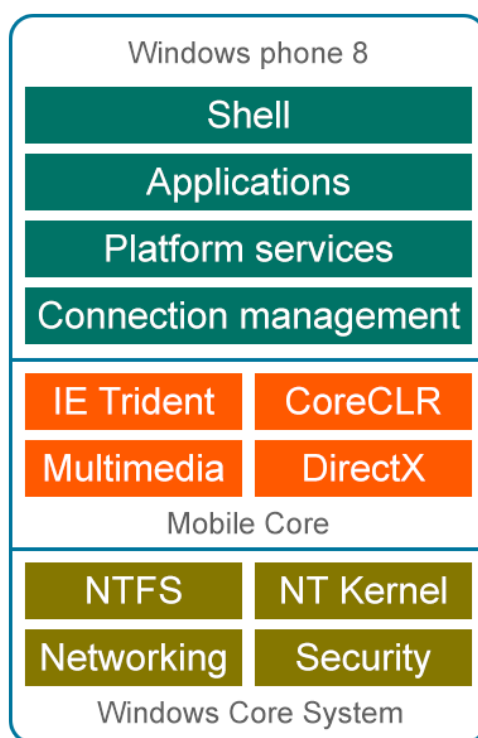
10.2.1 Vývoj verzí

Vývoj verze 7 se udál bez zásadních změn operačního systému. Jednalo se spíše o vývoj z hlediska integrovaných aplikací a zdokonalování jejich možností. Zásadnější změny proběhly u verze 7.5, kdy byla přidána podpora LTE sítí. Dále u verze 7.8, která změnila vzhled grafického rozhraní Metro do podoby aktuální verze 8. Změnil se pouze vzhled, nikoli jádro systému, které je stále Windows CE, tudíž s novější verzí není kompatibilní.

Verze 8 je přepracována na novou platformu NT, tudíž podporuje vyšší rozlišení displeje (1280x720 a 1280x768), více jádrové procesory (až 64). Jelikož vychází z platformy desktopových operačních systémů, je kompatibilní s aplikacemi systému Windows 8. Čerstvě představená verze 8.1 přináší minimální změny na úvodní obrazovce (viz. kapitola Grafické rozhraní). Změnilo se také uspořádání uzamykací obrazovky. Co se funkcí týče, byla přidána důležitá podpora VPN a hlasová asistentka Cortana [3].

10.3 Architektura Windows Phone 8

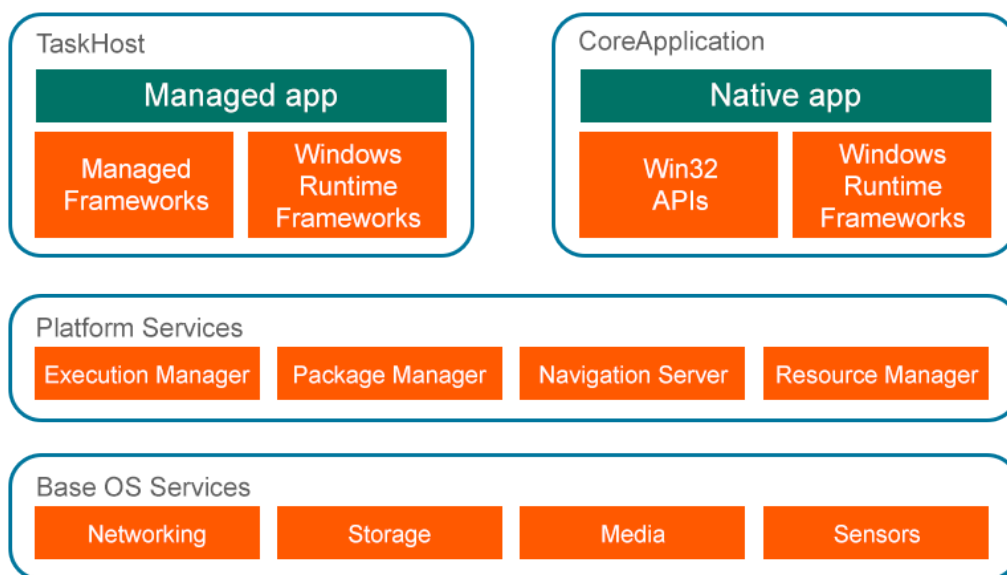
Základem platformy pro mobilní zařízení je sdílené jádro, jenž se skládá ze dvou částí - Windows Core System a Mobile Core. Windows Core System je společný pro desktopovou i pro mobilní platformu, obsahuje základní funkce systému. Mobile Core je přizpůsobený mobilním zařízením se zjednodušenou funkcionalitou. Společně zprostředkovávají sdílení funkcí a kódu mezi platformami. Nad sdíleným jádrem je umístěn samotný Windows Phone System, který obsahuje uživatelské rozhraní, služby, předinstalované aplikace a správu připojení.



Obrázek 10.1: Architektura Windows Phone 8.

10.4 Aplikační modely

Na sdíleném jádře je postaveno blokové schéma dvou aplikačních modelů. V levé části modelu je umístěn TaskHost, což představuje základní aplikační model XAML. Tento model se využívá již od Windows Phone 7. V pravé části modelu se nachází CoreApplication, která slouží aplikacím s grafickým rozhraním Direct3D přidána ve verzi Windows Phone 8.



Obrázek 10.2: Aplikační model Windows Phone 8.

10.5 Grafické rozhraní

Veškeré dosavadní verze Windows Phone využívají velmi podobné uživatelské rozhraní nazývané Metro. Rozhraní Metro spoléhá na jednoduchost, typografii a zaměření na obsah. Ikony aplikací, tak jako většina systému jsou orientovány ve čtvercích či obdélnících, proto jsou nazývány „dlaždice“. Dlaždice jsou dynamické, tudíž zobrazují informace o aplikaci v reálném čase.

10.6 Aplikace na pozadí systému

Microsoft zde volí kompromis mezi velkým množstvím spouštěných aplikací na pozadí systému, výkonnosti systému a spotřeby akumulátoru. Spousta aplikací závislých na systémových prostředcích a paměti při minimalizaci, znamenají zpomalení systému a zvýšenou spotřebu energie. Systém tak umožňuje spuštění procesů v pozadí systému pouze pro integrované aplikace. Ostatní aplikace ve vypnutém stavu nemají možnost

využívat systémové prostředky. Systém aplikacím znemožňuje práci i v případě, že je obrazovka pouze zamknutá. Komplikacím se zastavením aplikace při uzamčení obrazovky můžeme zabránit při vývoji aplikace, pod podmínkou zvýšení spotřeby akumulátoru.

Poznámka 10.1 S touto komplikací jsem se musel potýkat při vývoji mé aplikace, jelikož nutnou podmínkou příchodích požadavků je odezva zařízení. To se mi bohužel při vypnuté aplikaci splnit nepodařilo. Jedinou možností je v nastavení aplikace povolit nebo zakázat činnost aplikace při zamknuté obrazovce. To ovšem neznamená řešení celého problému, protože v tomto stavu aplikace by nebylo možné aplikaci vypnout a to nepříznivě ovlivní spotřebu baterie.

11 Vývoj aplikací

11.1 Nástroje pro vývoj

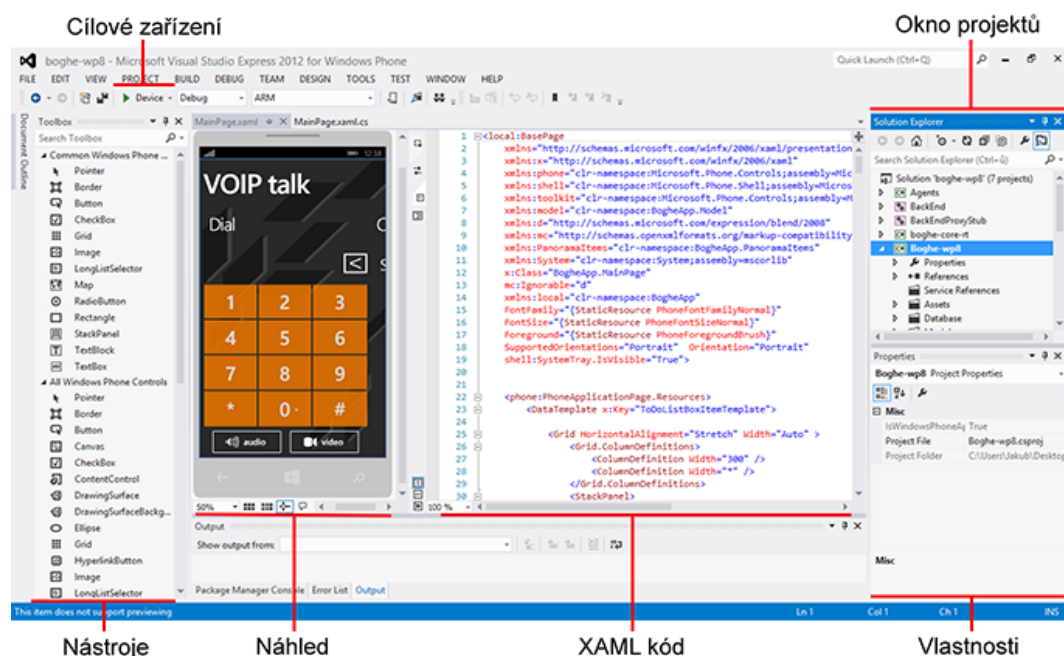
11.1.1 Windows Phone SDK

Nutným prostředkem pro vývoj aplikací je vývojový nástroj SDK. Existuje v několika verzích v závislosti na verzi OS, pro který chceme vyvíjet. Umožňuje instalaci do již předinstalovaného Visual Studia nebo samostatnou instalaci, kdy se instaluje celek Visual Studio for Windows Phone.

Každá verze SDK obsahuje potřebné komponenty pro vývoj aplikace. Pro potřeby Windows Phone 8 je potřeba SDK 8.0, které obsahuje nástroje pro tuto verzi operačního systému.

11.1.2 Představení vývojového prostředí

Na obrázku 11.1 vidíme rozpořádání, již nainstalovaného vývojového prostředí. Na pravé straně se nachází okno projektů a vlastností, které jsou obdobou standardního Visual Studia pro desktopové aplikace. V prostřední části je okno s XAML kódem a náhledem. Tyto dvě okna spolu úzce souvisí, jelikož změny provedené v jednom z nich se projeví v tom druhém. Vlevo na obrázku je vidět nástroje, které jsou odvozené z desktopových formulářových aplikací. Poslední důležitou záležitostí při tvorbě aplikací je spouštěcí tlačítka, kde při spuštění můžeme zvolit výstup sestavení aplikace. Na výběr máme několik předinstalovaných emulátorů, případně reálné zařízení připojené k počítači.



Obrázek 11.1: Nástroj Visual Studio určený k tvorbě aplikací pro Windows Phone.

11.2 Typy aplikací

Aplikace pro Windows Phone se rozdělují do tří skupin, podle použité struktury aplikace XAML, Direct3D a kombinace těchto dvou typů.

Typy aplikace	Popis	Programovací jazyk	Grafické rozhraní
XAML	Základní typ aplikace pro OS Windows Phone.	C#, Visual Basic	XAML
Direct3D	Využívá se pro graficky náročnější aplikace (hry). Zakládají se na aplikačním modelu z Windows, a tak dokáží plně využít potenciál základního hardwaru.	C/C++	Direct3D
kombinace	Aplikace, které jsou založeny na XAML struktuře s možností pracovat s Direct3D grafikou.	C#, Visual Basic, C/C++	XAML, Direct3D

Tabulka 11.1: Typy aplikací.

Poznámka 11.1 Pro vývoj mé aplikace jsem použil strukturu XAML, která je vhodná z důvodu absence složitější grafiky.

12 Aplikace VOIPtalk

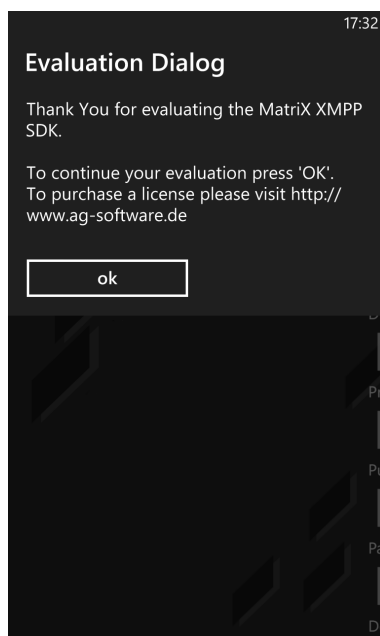
12.1 Použité frameworky a knihovny

12.1.1 MatriX XMPP

Knihovna je vytvořena společností AG-Software. Tato společnost nabízí prostředky pro komunikaci pomocí zpráv.

Jedná se o knihovnu pro tvorbu aplikací, využívající zprávy XMPP. Programovacím jazykem je C# .NET s využitím technologie Silverlight, to umožňuje knihovnu využít pro jakýkoliv .NET a Silverlight projekt. Knihovna obsahuje třídy a metody umožňující tvorbu aplikace pro přenos XMPP zpráv.

Poznámka 12.1 V mé aplikaci jsem použil zkušební verzi této knihovny, jelikož nebyly dostupné prostředky k zakoupení plné licence. Zkušební verze není po stránce funkcí nijak omezena. Knihovna není omezena ani co se týče doby použití. Jediným rozdílem je, že se při přihlášení k XMPP serveru objeví dialog 12.1, který podněcuje k zakoupení plné verze.



Obrázek 12.1: Screen aplikace s dialogem knihovny MatriX XMPP.

Poznámka 12.2 Cena licence pro jednoho vývojáře činí 1399 EUR na jeden rok, její obnovení 699 EUR [12].

12.1.2 Doubango

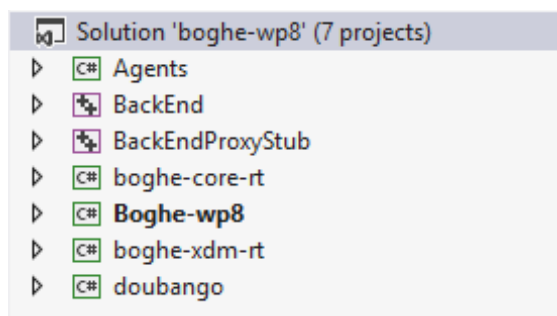
Framework vytvořený společností Doubango Telecom spadá pod open source licenci GNU, tudíž jej můžeme upravovat a dál jej šířit pod stejnou licenci. Zaměření této společnosti spočívá v open source projektech pro komunikace po síti.

Doubango je určeno pro tvorbu aplikací využívající VoIP technologii, konkrétně SIP protokol spolu s audio / video kódováním. Shrnuje spoustu různých platforem, od mobilních zařízení po standardní operační systémy. Například Android, Windows Phone, Windows, Mac OS, Linux.

VoIP je stále rozšířenější mezi mobilními platformami, proto Doubango přizpůsobilo framework mobilním zařízením s nižším výpočetním a paměťovým výkonem.

12.2 Struktura aplikace

Aplikace VOIPTalk je vytvořená na základě volně dostupné aplikace Boghe [1]. K vytvoření je využito otevřeného řešení aplikace této aplikace 12.2, které se skládá z pěti projektů a dvou dynamických knihoven Doubanga.



Obrázek 12.2: Struktura aplikace.

Stručný popis projektů a knihoven aplikace.

Agents Obsahuje třídy, které tvoří rozhraní pro připojení a ovládání VoIP procesů. Dále jsou zde třídy pro streamování a vykreslování videa.

BackEnd, BackEndProxyStub Dynamické knihovny Doubanga

boghe-core-rt Obsahuje příslušné služby a konfigurace služeb

Boghe-wp8 Spouštěcí projekt aplikace, který obsahuje grafické rozhraní a obsluhu grafických komponent. Také obsahuje databáze pro uložení kontaktů a zpráv.

boghe-xdm-rt Zde se nachází pouze jedna třída a enumerátor. Třída reprezentuje speciální názvy RCS a OMA. Enumerátor obsahuje stavy autorizace.

doubango Obsahuje metody stavových odpovědí při připojování k ústředně, nebo uskutečnění hovorů.

Poznámka 12.3 Bližší popis naleznete v dokumentaci na přiloženém CD.

12.3 Uživatelské rozhraní

Základní obrazovka aplikace je tvořena uživatelsky přívětivou panoramatickou obrazovkou, která usnadňuje a urychluje přesun mezi jednotlivými obrazovkami. Panorama je široké plátno, které přesahuje šířku obrazovky. Toto plátno je rozděleno na položky, kde každá položka představuje část plátna o šířce obrazovky. Přesun mezi jednotlivými obrazovkami se provádí jednoduchými pohyby do stran.

V případě mé aplikace je plátno rozděleno na čtyři položky. Jedná se o číselník, kontakty, zprávy a nastavení. Schéma uspořádání můžeme vidět na následujícím obrázku 12.3.

Tímto je aplikace ucelená v jedné obrazovce, tudíž nás nezdržuje přesun mezi různými celky aplikace. Graficky je oddělena pouze obrazovka hovoru a historie zpráv.



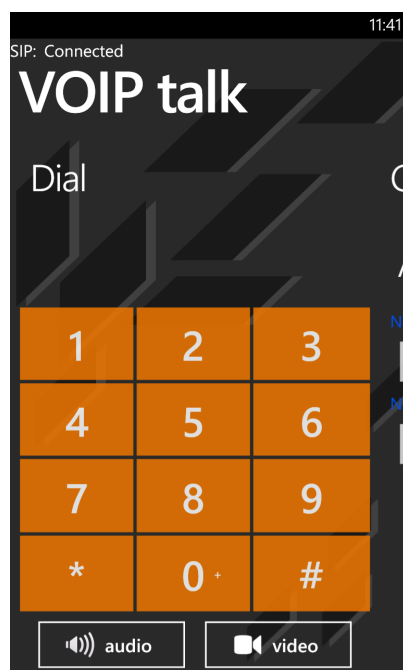
Obrázek 12.3: Schéma rozložení panoramatické obrazovky.

12.4 Uživatelská dokumentace

Tato část se věnuje ovládání aplikace VOIPtalk a potřebnému nastavení. Také jsou zde popsány funkce aplikace.

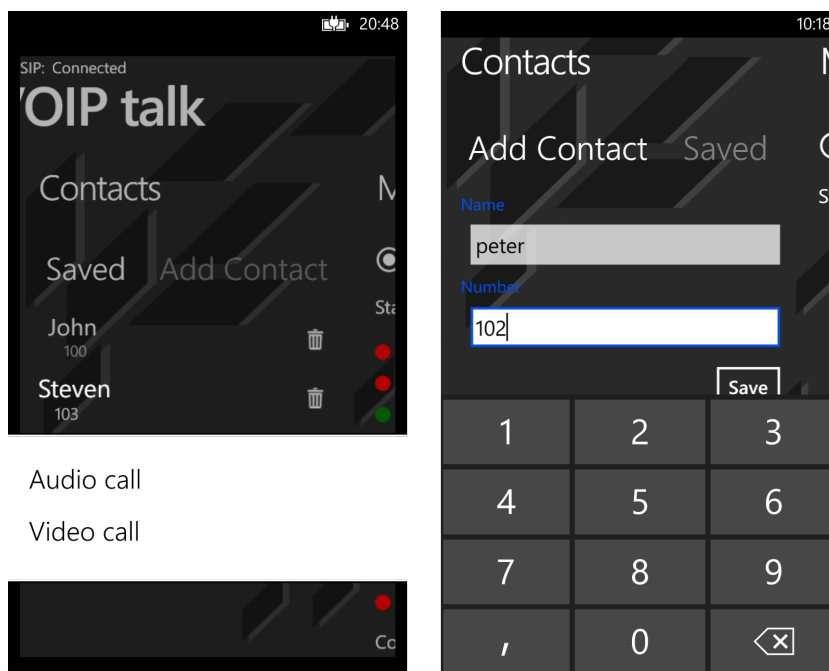
12.4.1 Ovládání aplikace

12.4.1.1 Číselník Po spuštění aplikace se objeví obrazovka 12.4 s číselníkem a ovládacími tlačítky hovoru. V levém horním rohu je zobrazen řádek se stavem připojení k SIP ústředně. Jakmile jsme řádně připojeni k ústředně, tlačítka vyvolání hovoru se aktivují. Poté můžeme vytočit číslo a uskutečnit hovor, dle stisku tlačítka audio nebo video.



Obrázek 12.4: Screen po spuštění aplikace.

12.4.1.2 Kontakty Ke kontaktům 12.5 se přesuneme tahem obrazovky na pravou stranu, ty jsou rozděleny do dvou částí. První z nich jsou uložené kontakty, vedle každého jména je ikona pro smazání kontaktu. Při dlouhém stisku jména kontaktu se zobrazí nabídka s možnostmi audio / video hovoru danému kontaktu. Obrazovka po pravé straně slouží pro přidávání kontaktů k vyplnění je zde pouze jméno a číslo.

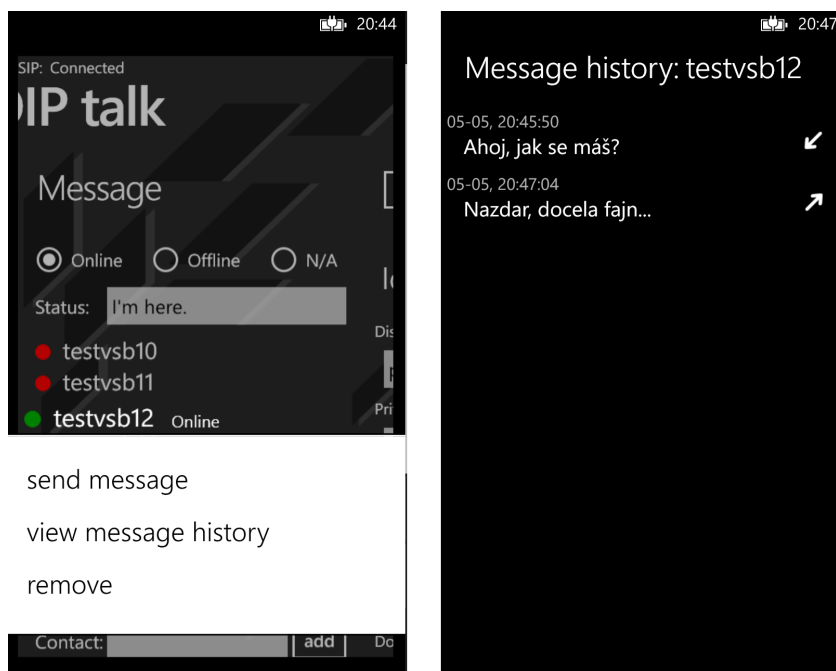


Audio call

Video call

Obrázek 12.5: Screen aplikace zobrazující práci s kontakty.

12.4.1.3 Zprávy Třetí obrazovka 12.6 obsahuje prostředí XMPP zpráv. Obrazovka obsahuje přihlášení k XMPP serveru a nastavení stavu. Po přihlášení se zobrazí kontakty, při delším stisku jména kontaktu se zobrazí možnost odeslání zprávy, zobrazení konverzace a odebrání kontaktu. Zobrazením konverzace se přesuneme na další obrazovku obsahující výpis komunikace s daným kontaktem.

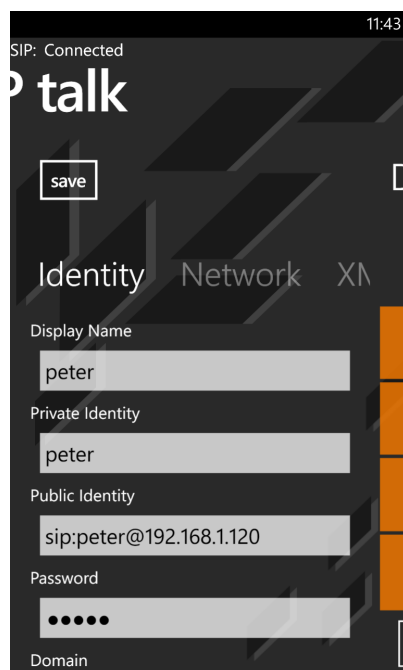


Obrázek 12.6: Screen aplikace zobrazující XMPP zprávy.

12.4.2 Nastavení aplikace

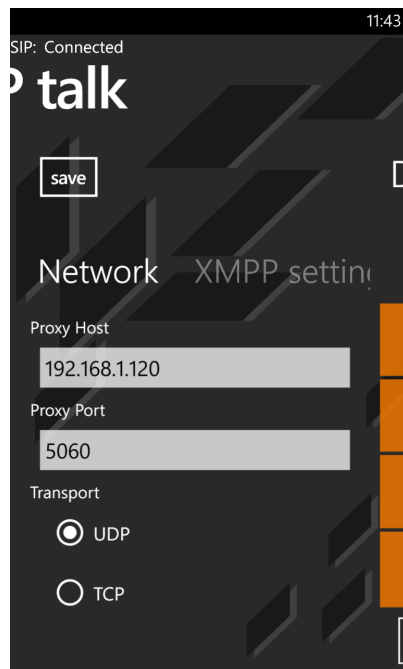
Nastavení aplikace je rozděleno na čtyři dílčí obrazovky. Nad těmito obrazovkami je tlačítko uložení, které uloží změny ze všech částí nastavení. Pro správnou funkci aplikace je nutné správně vyplnit všechny položky nastavení.

12.4.2.1 Identita V první části 12.7 máme možnost nastavit identitu sip účtu. Zde je nutné vyplnit zobrazované jméno, privátní identitu, veřejnou identitu, heslo a doménu.



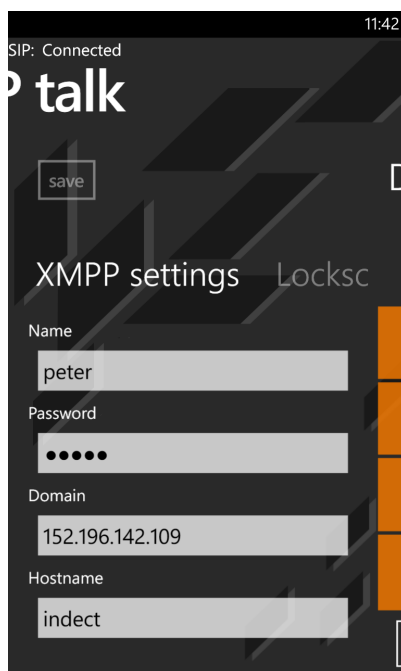
Obrázek 12.7: Screen aplikace s nastavením identity.

12.4.2.2 Sít' Další část 12.8 obsahuje nastavení sítě, která úzce souvisí s nastavením identity. Nastavuje se zde pouze proxy, port a transportní protokol.



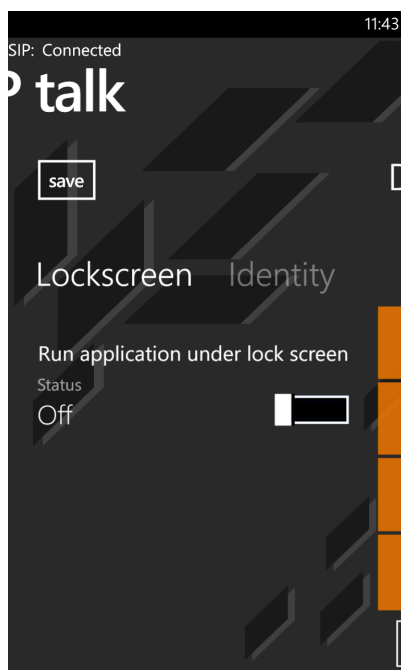
Obrázek 12.8: Screen aplikace s nastavením sítě.

12.4.2.3 XMPP nastavení Předposlední část 12.9 slouží k nastavení XMPP zpráv. Zde je nutné vyplnit uživatelské jméno a heslo, dále je tady nastavení domény a jména serveru. Pokud používáme server s doménovým jménem, pak je toto jméno potřeba nastavit do domény i do jména serveru.



Obrázek 12.9: Screen aplikace s nastavením XMPP zpráv.

12.4.2.4 Zamykací obrazovka Poslední část nesouvisí s nastavením komunikace, ale i přesto obsahuje podstatné nastavení této aplikace. Spouští se zde možnost přijímání hovorů a zpráv při zamknuté obrazovce 12.10.

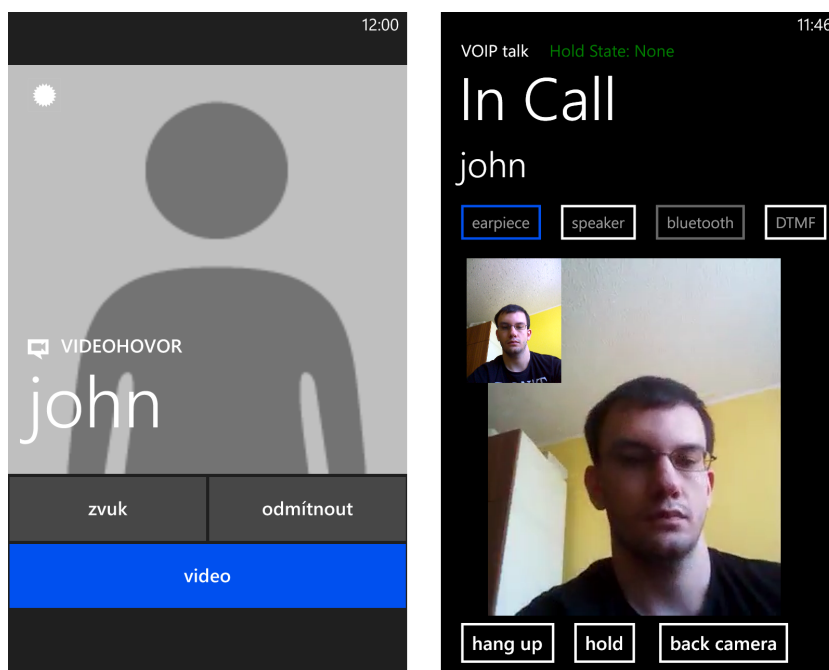


Obrázek 12.10: Screen aplikace s nastavením zamykání obrazovky.

12.4.3 Ovládání hovoru

Obrazovka přijetí hovoru obsahuje tři možnosti. Přijetí audio nebo video hovoru a odmítnutí hovoru.

Na obrazovce hovoru je zobrazen stav právě probíhajícího hovoru následuje číslo volaného, případně jeho jméno. Tlačítka v horní části slouží k ovládání výstupu zvuku (sluchátko, reproduktor a pomocí bluetooth), a také aktivace DTMF volby. Ve spodní části se nacházejí tlačítka pro ukončení a pozastavení hovoru. V případě video hovoru je zde navíc volba přepnutí mezi přední a zadní kamerou. V prostřední části se nacházejí obrazy obou uživatelů.



Obrázek 12.11: Screen aplikace zobrazující přijetí a průběh hovoru.

13 Testování aplikace

13.1 Apikační testování

Jelikož je aplikace vyvíjena pomocí Visual Studia, tak testování probíhalo zejména na vestavěných emulátorech s rozlišením 800x480, 1366x768 a 720p. Jediným reálným zařízením, které bylo k dispozici pro testovací účely je Nokia Lumia 920 s rozlišením 1280 x 768.

Aplikace vytvořené pro Windows Phone jsou poněkud striktní na verzi systému, z tohoto důvodu veškeré testovací prostředky pracovaly se stejnou verzí OS Windows Phone 8.

13.2 Testování zpráv

Po dohodě s vedoucím byl pro zprávy využit pouze XMPP protokol. Zprávy jsem testoval pomocí dvou serverů. První z nich je Openfire 3.8.2 umístěný ve školní síti, jako druhý jsem využil veřejně přístupný server www.jabbim.cz.

13.2.1 Server Openfire

slouží pro přenos XMPP zpráv v reálném čase. Je open source pod licencí Apache. Vzhledem k jeho snadné instalaci je často využíván pro přenos zpráv v lokálních sítích. Snadný po stránce nastavení, které je realizováno pomocí přehledného webového rozhraní. Pro testování jsou na serveru vytvořeni klienti se stejnými jmény a hesly jako byly použity u testování hovorového spojení SIP.

13.2.2 Server Jabbim.cz

je veřejně dostupný, postačuje se registrovat u tohoto poskytovatele. Co se týče komunikace je server stejný jako Openfire, ale nemáme možnost nastavení serveru, jelikož již nejsme v roli správce.

13.3 Testování SIP spojení

Z pohledu SIP spojení bylo testování prováděno spolu s ústřednou Asterisk, kterou jsem provozoval virtuálně na vlastním počítači.

Asterisk je volně dostupné řešení postavené na systému Linux / Unix. Je určeno pro podnikové sítě, umožňuje VoIP komunikaci skrz telekomunikační protokoly, nejčastěji SIP a H.323. Pomocí tohoto řešení můžeme přenášet audio a video data v reálném čase.

Pro testování je vytvořeno na serveru pár uživatelských účtů

- Uživatelské jméno: john, Heslo: john, Číslo: 100
- Uživatelské jméno: james, Heslo: james, Číslo: 101
- Uživatelské jméno: peter, Heslo: peter, Číslo: 102

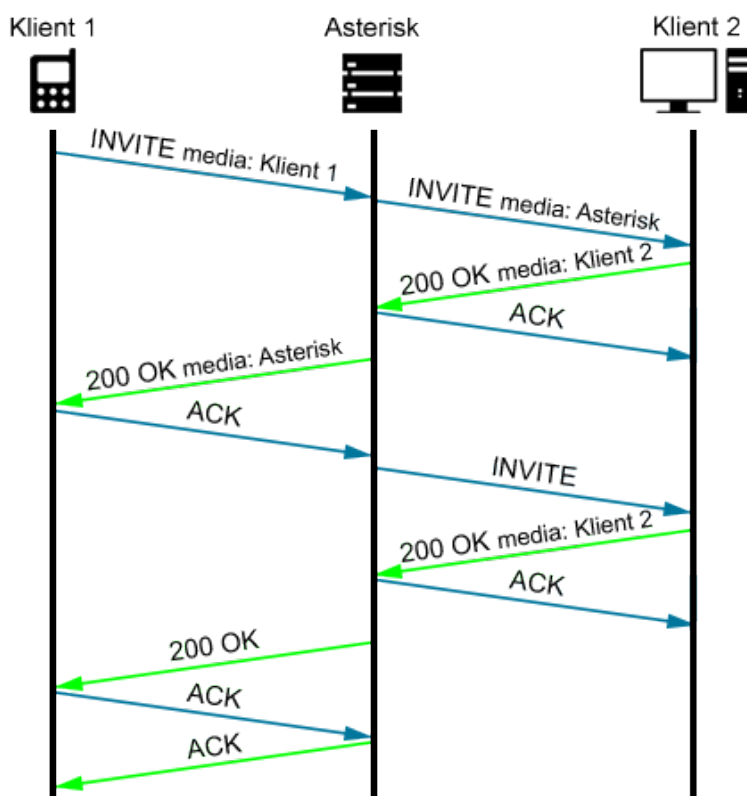
13 TESTOVÁNÍ APLIKACE

- Uživatelské jméno: steven, Heslo: steven, Číslo: 103

Testování probíhalo mezi aplikací VOIPtalk na telefonu Nokia a jako druhý klient byl použit cizí volný software v počítači nebo v telefonu se systémem Android. Jednalo se o aplikace Zoiper a IMSdroid.

13.3.1 Spojení dvou klientů

pomocí Asterisku představuje kroky vyobrazené na schématu 13.1. Nejprve zařízení provede požadavek na ústřednu Asterisk, která vyhledá a zkontaktuje dotazovaného účastníka. Poté předá informace o volaném zpátky na zařízení žádající o spojení, následná komunikace již probíhá mezi dvěma koncovými body bez přítomnosti serveru. Touto formou komunikace se snižuje zatížení serveru a spoždění hovoru [14].



Obrázek 13.1: Spojení dvou koncových zařízení pomocí ústředny Asterisk.

14 Závěr

Cílem mé bakalářské práce bylo vytvořit VoIP aplikaci založenou na SIP protokolu s podporou pro textovou komunikaci. Aplikace je určena pro operační systém Windows Phone. Vycházel jsem z již vytvořené aplikace Boghe, která je určena pro operační systém Windows Phone 8. Aplikace měla vytvořenou funkcionalitu hovoru, ale byla velmi jednoduše tvořená, krom hovorové části neposkytovala žádné jiné funkce.

Na základě výše zmíněné aplikace jsem vytvořil aplikaci VOIPtalk, do které byly přidány použité kodeky, textová komunikace na základě protokolu XMPP a databáze pro uložení potřebných dat.

Po dohodě s vedoucím bakalářské práce byl vynechán SOAP protokol a textová komunikace pomocí SIP protokolu. SOAP měl sloužit pro ovládání ústředny, ale od toho řešení bylo upuštěno. Textová komunikace prostřednictvím protokolu SIP, byla vynechána z důvodu implementování rozšířenější XMPP komunikace.

Testování hovorové komunikace bylo uskutečněno s ústřednou Asterisk, která byla umístěna ve školní síti a virtuálně na mém počítači. Dále bylo potřeba otestovat textovou komunikaci XMPP, to bylo provedeno pomocí Openfire a Jabbim serveru.

Další vývoj aplikace je vhodný zejména z pohledu přenesení na čerstvě vydanou verzi operačního systému Windows Phone 8.1, která umožňuje aplikaci spustit procesy na pozadí systému. To zajistí možnost příchozích hovorů a zpráv při vypnuté aplikaci. Dosud aplikace tuto podstatnou funkci neobsahuje, jelikož je omezená ze strany dosavadní verze systému.

15 Literatura

- [1] Boghe: Building Boghe wp8 [online]. [cit. 2014-04-20]. Dostupné z: https://code.google.com/p/boghe/wiki/Building_Boghe_v2_wp8
- [2] Webopedia: The History of Microsoft Operating Systems. [online]. [cit. 2014-04-20]. Dostupné z: http://www.webopedia.com/DidYouKnow/Hardware_Software/history_of_microsoft_windows_operating_system.html
- [3] Windows Mobile mania: Windows Phone 8.1 (Blue) – Velký přehled novinek a vylepšení. [online]. [cit. 2014-04-20]. Dostupné z: <http://wmmania.cz/clanky/obecne/windows-phone-blue-prehled-novinek>
- [4] MSDN: The H.323 Standard [online]. [cit. 2014-04-20]. Dostupné z: [http://msdn.microsoft.com/en-us/library/ms709083\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/ms709083(v=vs.85).aspx)
- [5] M. Vozňák: Signalizační protokoly v NGN. [cit. 2014-05-02]. Dostupné z: http://homel.vsb.cz/~voz29/ss/ss_13pr.pdf
- [6] M. Vozňák: Kvalita hovoru v prostředí VoIP. [cit. 2014-05-02]. Dostupné z: http://homel.vsb.cz/~voz29/files/voz_71.pdf
- [7] RTP: A Transport Protocol for Real-Time Applications. [cit. 2014-05-02]. Dostupné z: <http://www.ietf.org/rfc/rfc3550.txt>
- [8] Úvod do VoIP. [cit. 2014-05-03]. Dostupné z: http://www.mbddata.cz/uvoddovoip.htm#_Protokol_RTP_a_RTPC
- [9] XMPP standards foundation: XMPP Technologies Overview. [online]. [cit. 2014-04-21]. Dostupné z: <http://xmpp.org/about-xmpp/technology-overview>
- [10] XMPP standards foundation: XMPP Extensions. [online]. [cit. 2014-04-21]. Dostupné z: <http://xmpp.org/xmpp-protocols/xmpp-extensions>
- [11] HW.cz: DTMF - fámy a skutečnost. [online]. [cit. 2014-04-23]. Dostupné z: <http://www.hw.cz/teorie-a-praxe/dokumentace/dtmf-famy-a-skutecnost.html>
- [12] AG-Software: Purchase. [online]. [cit. 2014-04-24]. Dostupné z: <http://www.ag-software.net/matrix-xmpp-sdk/purchase/>
- [13] Využití webových služeb a protokolu SOAP při komunikaci: Kapitola 4. Komunikační infrastruktura. [online]. [cit. 2014-05-03]. Dostupné z: <http://www.kosek.cz/diplomka/html/websluzby.html>
- [14] Asterisk: SIP Direct Media Reinvite Glare Avoidance. [online]. [cit. 2014-04-25]. Dostupné z: <https://wiki.asterisk.org/wiki/display/AST/SIP+Direct+Media+Reinvite+Glare+Avoidance>

A Přílohy na CD

Adresářová struktura přiloženého CD:

Aplikace – Adresář obsahuje instalační soubor aplikace.

Kompletní projekt – Obsahuje kompletní spustitelný projekt aplikace, včetně všech knihoven a návodu na spuštění projektu.

Dokumentace – Obsahuje původní dokumentaci aplikace, mé práce a Douabango frameworku.